Rendu basé image avec contraintes sur les gradients

Grégoire Nieto, Frédéric Devernay, James L. Crowley

Rendu basé image avec contraintes sur les gradients

Grégoire Nieto Frédéric Devernay James Crowley

INRIA Grenoble Rhône-Alpes - équipe IMAGINE et PRIMA
LIG Laboratoire d'informatique de Grenoble

gregoire.nieto@inria.fr

Résumé

Le rendu basé image consiste à générer un nouveau point de vue à partir d'un ensemble de photos d'une scène. On commence en général par effectuer une reconstruction 3D approximative de la scène, utilisée par la suite pour mélanger les images sources et ainsi obtenir l'image souhaitée. Malheureusement, les discontinuités dans les poids des images sources, dues à la géométrie de la scène ou au placement des caméras, causent des artefacts visuels dans la vue résultante. Dans cet article nous montrons qu'une façon d'éviter ces artefacts est d'imposer des contraintes supplémentaires sur le gradient de l'image synthétisée. Nous proposons une approche variationnelle suivant laquelle l'image cherchée est solution d'un système linéaire résolu de façon itérative. Nous testons la méthode sur plusieurs jeux de données multi-vues structurés et non-structurés, et nous montrons que non seulement elle est plus performante que les méthodes de l'état de l'art, mais elle élimine aussi les artefacts créés par les discontinuités de visibilité.

Mots Clef

Rendu basé image, reconstruction 3D, imagerie computationelle

Abstract

Multi-view image-based rendering consists in generating a novel view of a scene from a set of source views. In general, this works by first doing a coarse 3D reconstruction of the scene, and then using this reconstruction to establish correspondences between source and target views, followed by blending the warped views to get the final image. Unfortunately, discontinuities in the blending weights, due to scene geometry or camera placement, result in artifacts in the target view. In this paper, we show how to avoid these artifacts by imposing additional constraints on the image gradients of the novel view. We propose a variational framework in which an energy functional is derived and optimized by iteratively solving a linear system. We demonstrate this method on several structured and unstructured multi-view datasets, and show that it numerically outperforms state-of-the-art methods, and eliminates artifacts that result from visibility discontinuities.

Keywords

Image-Based Rendering, 3D Reconstruction, Computational Photography

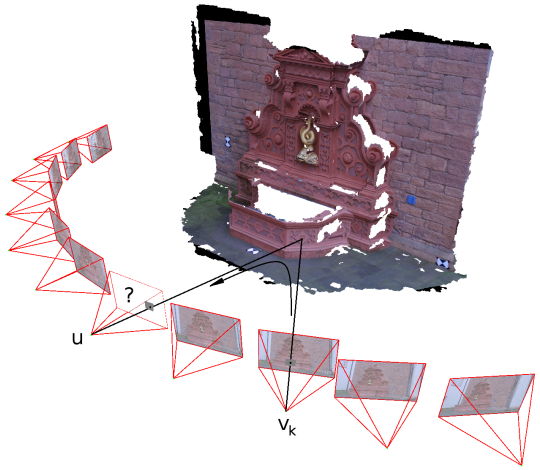

FIGURE 1 – La reconstruction 3D de la scène permet de mettre en correspondance la vue cible u avec les vues sources v_k.

1 Introduction

Le rendu basé image consiste à générer un nouveau point de vue à partir d'un ensemble de photos d'une scène. On commence en général par effectuer une reconstruction 3D approximative de la scène, appelée *proxy géométrique*, qui est ensuite utilisée pour établir des correspondances entre les vues sources et la vue à synthétiser (Figure 1). Enfin des vues projetées sont mélangées pour obtenir l'image finale. Le récent travail de Pujades *et al*. [26] propose une formulation bayésienne du problème de rendu basé image, construite sur le travail précédent de Wanner et Goldluecke [13,33]. Ils ont montré que le poids de chaque image source dans le mélange final pouvait se déduire formellement des propriétés de la caméra, du contenu de l'image, et de la précision du *proxy géométrique*, amenant une nouvelle formalisation des heuristiques de mélange proposées

initialement par Buehler *et al*. [4]. La plupart des « propriétés désirables qu'un algorithme idéal de rendu basé image devrait avoir » [4] eurent alors une explication formelle, sauf la propriété de *continuité*. Par conséquent, les discontinuités dans les poids des images sources, dues principalement à l'estimation de la géométrie de la scène ou au placement des caméras, créent des artefacts visuels dans la vue synthétisée.

Dans cet article, nous montrons qu'une façon d'éviter ces artefacts est d'imposer des contraintes supplémentaires sur le gradient de l'image synthétisée. Ces contraintes viennent d'une simple observation : les contours d'image dans la vue cible doivent aussi être des contours dans les images source où ces parties sont visibles. Une fonctionnelle d'énergie similaire à celle de Pujades *et al*.est développée, composée de l'habituel terme sur les données (*data term*) et un terme de régularisation (*smoothness term*), mais le terme sur les données contient un terme additionnel qui prend en compte les contraintes sur les gradients.

Dans la section 3, nous décrivons comment nous estimons les transformations d'images pour les projeter sur la vue cible. Nous développons une formulation variationnelle du modèle de rendu basé image dans la section 4. Les hypothèses de discrétisation et d'implémentation de notre méthode sont décrites dans la section 5. Dans la section 6, nous testons la méthode sur plusieurs jeu de données multi-vues structurés et non-structurés, et nous montrons que non seulement elle est plus performante que les méthodes de l'état de l'art, mais elle élimine aussi les artefacts créés par les discontinuités de visibilité. Nous concluons que tenir compte à la fois de l'intensité et du gradient dans les méthodes de rendu basé image apporte une solution élégante au renforcement de la propriété de *continuité* initialement énoncée par Buehler *et al*.

2 Travaux antérieurs

Les techniques de rendu basé image ont été décrites et classifiées par Shum *et al*. [27]. La plupart des méthodes de l'état de l'art [6, 18, 20, 21, 28, 35] utilisent une reconstruction de la géométrie de la scène plus ou moins précise, appelée géométrie intermédiaire ou *proxy géométrique*. Ortiz-Cayon *et al*. [24] proposent de segmenter les images selon les zones où chaque algorithme est susceptible de produire le moins d'artefacts possible. Toutes ces techniques sont inspirées par *Unstructured Lumigraph Rendering* [4], qui effectue un mélange des k plus proches vues, pondérées par les angles et les distances à la vue cible. La technique de rendu de Davis *et al*. [8] va plus loin en renforçant la continuité des poids du mélange, supprimant ainsi la plupart des artefacts temporels. Néanmoins, les poids de mélange sont encore calculés d'après des règles heuristiques et le choix des caméras pour le rendu est totalement arbitraire.

L'idée clé de se débarrasser des heuristiques est apportée par Wanner *et al*. [33] qui proposent une formulation bayésienne du rendu basé image fondée sur un modèle physique de formation de l'image. Les poids des contributions de chaque vue source dans le mélange final est automatiquement déduit des équations mathématiques en dérivant une fonctionnelle énergie. Pujades *et al*. [26] sont allés plus loin en intégrant l'incertitude de la géométrie dans le formalisme bayésien. Ils obtiennent de nouveaux poids qui favorisent à la fois la *cohérence épipolaire* et la *déviation angulaire minimale*, critères évoqués dans Buehler *et al*. [4] pour décrire l'algorithme de rendu basé image idéal. Ils n'apportent cependant pas d'éléments pouvant satisfaire pleinement le principe de *continuité*, en particulier près des bords du champ de vue de chaque caméra source. Nous montrons que l'introduction d'un terme additionnel dans la fonctionnelle énergie ajoute de nouvelles contraintes sur le gradient de l'image cherchée, empêchant l'apparition de hautes fréquences dues au mauvais conditionnement du système et apportant une solution élégante au problème de conditionnement.

La fusion d'images dans le domaine du gradient a reçu beaucoup d'intérêt ces dernières années, en commençant par l'article phare de Perez *et al*. [25], pour des applications dans l'édition d'images [22], l'*inpainting* [19] ou les panoramas [1, 36]. Le travail le plus proche du notre en rendu basé image est probablement celui de Kopf *et al*. [18], qui génère effectue un rendu de gradient, suivi d'une intégration pour produire la couleur de l'image. Néanmoins cette méthode se limite à interpoler entre deux vues sources, et elle ne s'attelle pas à la configuration générique, dans laquelle les points de vue sources sont très différents et non structurés, ce que nous proposons de faire.

3 Reconstruction 3D

La plupart des méthodes de rendu basé image utilisent une reconstruction 3D approximative de la scène appelée *proxy géométrique*. Nous avons opté pour une représentation en cartes de profondeur car elles sont un bon compromis entre précision et exhaustivité de reconstruction. En effet, la reconstruction d'un nuage de point à l'aide d'un algorithme de l'état de l'art [11] est économe en données et très précise mais les données sont éparses. D'autre part, si une reconstruction de surface [9,16] est faite à partir du nuage de point dans le but de densifier les correspondances entre les vues, la précision de la géométrie diminue. Les cartes de profondeur offrent en outre l'avantage d'établir immédiatement les correspondances entre tout point $\mathrm{x}_m = (x_m, y_m, 1)^\intercal$ d'une vue source v_k son projeté $\mathrm{x}_p = (x_p, y_p, 1)^\intercal$ sur la vue cible u. Nous appelons τ_k une telle transformation :

$$\begin{array}{rccc} \tau_k : & \Omega_k & \rightarrow & \Gamma \\ & \mathrm{x}_m & \mapsto & \mathrm{x}_p \end{array} \tag{1}$$

3.1 Calibration des caméras

Nous utilisons une partie du logiciel de reconstruction 3D multi-vues MVE [10]. Dans un premier temps, nous corrigeons la distorsion des caméras et les calibrons à l'aide d'openMVG [23]. Le modèle sténopé est alors choisi pour

FIGURE 2 – La carte de profondeur de la vue cible, obtenue par projection de quads depuis les vues sources. Images de la base *fountain*.

représenter les caméras : une matrice K_k de paramètres intrinsèques, ainsi que la rotation R_k et la translation t_k permettant le changement en coordonnées monde/caméra.

3.2 Correspondances par pixel

Pour chaque vue k, une carte de profondeur est estimée en utilisant l'algorithme de stéréo multi-vues [12]. Les profondeurs obtenues h sont radiales – distances euclidiennes entre un point 3D de la scène et le centre de la caméra. Nous les convertissons en profondeurs orthogonales :

$$z(\mathrm{x}_m) = \frac{h(\mathrm{x}_m)}{\|K_k^{-1}\mathrm{x}_m\|}. \tag{2}$$

De la même façon, la dérivée spatiale $h_\mathrm{x} = \frac{\partial h}{\partial \mathrm{x}}$ donnant l'orientation de la surface peut être convertie en

$$z_\mathrm{x}(\mathrm{x}_m) = \frac{1}{\|K_k^{-1}\mathrm{x}_m\|}(h_\mathrm{x}(\mathrm{x}_m) - \frac{(K_k^{-1}\mathrm{x}_m)^\intercal K_k^{-1}[0]}{\|K_k^{-1}\mathrm{x}_m\|^2} h(\mathrm{x}_m)) \tag{3}$$

où $K_k^{-1}[0]$ représente la première colonne de K_k^{-1}.
Chaque carte de profondeur est filtrée par un filtre bilatéral [17] dans le but de combler les trous.
Les transformations d'images τ_k sont calculées en projetant sur la vue cible le point 3D estimé par la carte de profondeur de la vue source :

$$\tau_k(\mathrm{x}_m) = N_e(K_u R_u(R_k^\intercal(z(\mathrm{x}_m)K_k^{-1}\mathrm{x}_m) - t_k) + t_u) \tag{4}$$

où N_e est la normalisation euclidienne, qui consiste à diviser un vecteur en coordonnées homogènes par sa dernière composante, ici la profondeur orthogonale du point 3D vu depuis u.

3.3 Gestion des occultations

La gestion de la visibilité se fait en deux temps. D'abord, les pixels des vues sources sont marqués invalides si leur projeté par τ_k se situent hors des bords de l'image de la vue cible. Ensuite, la carte de profondeur z_u de la vue cible est estimée pour traiter les occultations inverses (Figure 2). À partir de chaque pixel x_m de chaque vue source v_k un *quad* (quadrilatère 3D) est créé à la distance estimée $z(\mathrm{x}_m)$ et orienté par $z_\mathrm{x}(\mathrm{x}_m)$. Il est ensuite projeté sur la vue cible u en accumulant un z-buffer pour ne retenir que les profondeur les plus proches. Le test de visibilité s'effectue en comparant la distance du point 3D reconstruit à la vue cible avec la profondeur estimée précédemment : si la différence se situe au-delà d'un certain seuil – fixé arbitrairement – alors le pixel x_m est marqué comme non visible depuis u.

3.4 Contributions des vues sources

Les poids des contributions de chaque vue source dans le rendu sont de deux sortes [26]. D'une part les poids de déformation, donnés par la jacobienne de la transformation τ_k, pénalisent les vues qui observent la surface de biais ou qui se situent loin de la vue cible. La jacobienne se calcule à partir des matrices des caméras, des profondeurs et des normales estimées :

$$\frac{\partial \tau_k}{\partial \mathrm{x}_m}(\mathrm{x}_m) = J_e K_u R_u R_k^t K_k^{-1} \begin{pmatrix} h_x x_m + z & h_y x_m \\ h_x y_m & h_y y_m + z \\ h_x & h_y \end{pmatrix} \tag{5}$$

Les poids de géométrie, garantissant la *déviation angulaire minimale*, c'est-à-dire pénalisant les caméras formant un angle trop grand avec la vue cible, découlent de l'incertitude de la géométrie estimée $\sigma_{g,k}^2$ et de la variance du bruit du capteur $\sigma_{s,k}^2$. Chaque vue source k est ainsi pondérée par $\omega_k(u) = (\sigma_{s,k}^2 + \sigma_{g,k}^2(u))^{-1}$ avec $\sigma_{g,k}^2(u) = \sigma_{z,k}^2(b \star (\frac{\partial \tau_k}{\partial z}^t \nabla u \circ \tau_k))^2$, et ∇u le gradient de la solution courante u. Les dérivées $\frac{\partial \tau_k}{\partial z}$ sont données par la formule

$$\frac{\partial \tau_k}{\partial z}(\mathrm{x}_m) = J_e K_u R_u R_k^t K_k^{-1} \mathrm{x}_m \tag{6}$$

4 Une formulation variationnelle du rendu basé image

La deuxième étape de notre méthode est une approche variationnelle pour synthétiser une nouvelle image optimale $u : \Gamma \to \mathbb{R}$ au point de vue cible à partir d'images sources $v_k : \Omega_k \to \mathbb{R}$. Pour plus de clarté, les valeurs des images sont prises comme étant des scalaires, mais il est aisé de généraliser aux images couleurs.

4.1 Modèle de formation de l'image

Comme on suppose en général dans la littérature sur la super-résolution [2, 15], la valeur d'intensité $v_k(x_m)$ d'un point x_m dans l'image source k peut s'écrire comme la convolution de l'image cible avec la fonction d'étalement du point (FEP) ou noyau de flou, notée b. Étant donnée une image idéale u au point de vue ciblé, définie sur le domaine Γ, et une transformation τ_i des points de Ω_k dans Γ, si nous mettons de côté les occultations pour le moment, l'intensité de l'image observée peut s'écrire comme

$$v_k(\mathrm{x}_m) = \int_{\Omega_k} u \circ \tau_k(\mathrm{x})\, b(\mathrm{x} - \mathrm{x}_m)\, \mathrm{dx}, \tag{7}$$

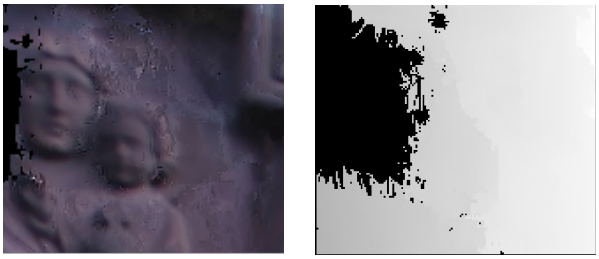

FIGURE 3 – Les discontinuités des transformations τ_k et des poids ω_k provoquent des artefacts. A gauche : une vue du jeu de données de Strecha [29] estimée avec l'énergie donnée par l'équation 8. A droite : une transformation τ_k présentant des discontinuités dues à la visibilité à cet endroit de l'image.

ou plus simplement $v_k = b * (u \circ \tau_k)$.

4.2 Estimation du maximum *a posteriori*

Le but de toute approche variationnelle est d'estimer une image u à partir des données $(v_k^*)_{k\in[1..K]}$, où K est le nombre de vues sources. L'estimateur de u doit maximiser la probabilité *a posteriori* d'observer u sachant nos données en entrée et la probabilité *a priori* de u. On peut montrer que c'est équivalent à minimiser l'énergie

$$E(u) = E_{\text{color}}(u) + \lambda E_{\text{prior}}(u), \tag{8}$$

où E_{color} est le terme portant sur les données. E_{prior}, souvent appelé terme de régularisation, vient de la probabilité *a priori* de l'image, et λ permet de lisser la solution finale en prévenant l'apparition de hautes fréquences. Dans notre travail, nous utilisons un *a priori* de variation totale [14], $E_{\text{prior}}(u) = \int_\Gamma |\nabla u|$, qui a quelques avantages par rapport à d'autres *a priori* d'image plus complexes [7, 30] : il préserve les bords abruptes et contours de l'image, et il est convexe. Une preuve de convergence est fournie par Chambolle [5].
E_{color} est donné par le terme de vraisemblance [26]. Cela prend en compte dans quelle mesure la solution courante est en adéquation avec les données :

$$E_{\text{color}}(u) = \sum_{k=1}^{K} \frac{1}{2} \int_{\Omega_k} \omega_k(u) \left((b * (u \circ \tau_k) - v_k^*)\right)^2 \,\mathrm{dx}, \tag{9}$$

4.3 Ajout du terme portant sur le gradient de l'image

Puisque la géométrie et la visibilité peuvent être discontinues et bruitées, le terme $\omega_k(u)$ dans (9) et les τ_k peuvent être aussi très bruités, ce qui résulte en des artefacts dans la solution finale qui apparaissent comme de faux bords ou textures (Figure 3). La méthode de rendu basé image devrait empêcher ces contours d'apparaître : en fait, un contour synthétisé dans l'image solution devrait également être présent dans les images sources, là où ces parties de la scène sont visibles.

Pour renforcer cette propriété, nous ajoutons un terme supplémentaire $E_{\text{grad}}(u)$ à l'énergie précédente (9), qui force la solution courante à se rapprocher des données dans le domaine du gradient :

$$\begin{aligned} E_{\text{grad}}(u) &= -\log \mathrm{p}(\nabla v_0 \dots \nabla v_{K-1} | \nabla u) && (10)\\ &= -\sum_{k=0}^{K-1} \log \mathrm{p}(\nabla v_k | \nabla u) && (11)\\ &= \int_{\Omega_k} (\nabla v_k - \nabla v_k^*)^2 \,\mathrm{dx} && (12)\\ &= \int_{\Omega_k} (\nabla (b * (u \circ \tau_k)) - \nabla v_k^*)^2 \,\mathrm{dx}. && (13) \end{aligned}$$

Ce terme permet en outre d'ajouter de nouvelles contraintes au système qui est alors mieux conditionné.
Trouver u qui minimise cette énergie est alors équivalent à résoudre l'équation de Laplace :

$$\Delta((b * (u \circ \tau_k) - v_k^*) = 0, \tag{14}$$

où $\Delta = \nabla.\nabla$ représente le laplacien de l'image. On en déduit aisément la différentielle de la fonctionnelle :

$$\mathrm{d}E_{\text{grad}}(u) = (|\frac{\partial \tau_k}{\partial z}|)^{-1}\, \bar{b} * (\Delta(b * (u \circ \tau_k)) - \Delta v_k^*)) \circ \beta_k. \tag{15}$$

Les β_k sont les transformations inverses qui apparaissent à cause du changement de variable dans l'intégrale. $\bar{b}$ est l'adjoint de la FEP b. Les transformations τ_k sont celles qui ont été estimées auparavant, et manquent ainsi de précision. Cette incertitude a un effet néfaste sur le calcul de $\Delta(b * (u \circ \tau_k))$. Par conséquent, nous choisissons de calculer le laplacien de u d'abord, puis de le transformer dans le domaine Ω_k. Sous l'hypothèse que les transformations τ_k sont localement linéaires, on peut négliger leurs dérivés au deuxième ordre et obtenir

$$\Delta(b*(u\circ\tau_k)) = b*\left(\frac{\partial \tau_k}{\partial x}^t H_u(\mathrm{x}) \frac{\partial \tau_k}{\partial x} + \frac{\partial \tau_k}{\partial y}^t H_u(\mathrm{x}) \frac{\partial \tau_k}{\partial y}\right), \tag{16}$$

où $H_u = \frac{\partial \nabla u}{\partial \mathrm{x}}$ est la hessienne de u.
Les cartes de profondeur mal estimées causant de fortes discontinuités dans les correspondances entre les vues, la hessienne peut être très instable. Pour l'implémentation et dans ce cas seulement, nous supposons que $\tau_k(\mathrm{x}) \approx \mathrm{x} + d$, de telle façon que

$$\Delta(b * (u \circ \tau_k)) = b * (\mathrm{trace}(H_u(\mathrm{x})) \circ \tau_k) = b * (\Delta u \circ \tau_k). \tag{17}$$

La forme finale de l'énergie à minimiser est donc

$$E(u) = \alpha E_{data}(u) + \gamma E_{grad}(u) + \lambda E_{prior}(u). \tag{18}$$

Nous minimisons la fonctionnelle (18) via FISTA (*Fast Iterative Shrinkage Thresholding Algorithm*) [3].

5 Discrétisation et approximation

5.1 La fonction d'étalement du point

La fonction d'étalement du point (FEP) $b : \Omega_k \to [0,1]$ est une densité de probabilité qui peut être transformée en $b_k : \Gamma \to [0,1]$ par le changement de variable $\mathrm{x}' = \tau_k(\mathrm{x})$ de telle manière que

$$b_k(\mathrm{x}') = b(\mathrm{x}) = |J_{\tau_k}(\mathrm{x})|\, b(\tau_k(\mathrm{x})). \tag{19}$$

Soit $x_p = \tau_k(\mathrm{x}_m) \in \Gamma$. Alors, par changement de variable à l'intérieur de l'intégrale, on obtient :

$$v_k(\mathrm{x}_m) = \int_\Gamma u(\mathrm{x}') b_k(\mathrm{x}' - \tau_k(\mathrm{x}_m))\, \mathrm{dx}' \tag{20}$$

Il y a plusieurs façons de calculer la transformée de la FEP, selon comment on la représente dans Ω_k. On utilise le plus souvent une gaussienne 2D de moyenne x_m et de covariance Σ. Puisque la gaussienne a un support infini, cela est difficile à implémenter en pratique. Une modélisation plus simple consiste à faire l'hypothèse d'un pixel carré et uniformément sensible à la lumière sur toute sa surface. Ainsi, la FEP est une fonction de densité uniforme carrée. En notant A l'aire d'un pixel centré sur $(0,0)$ dans une vue v_k, on a :

$$b(x,y) = \begin{cases} \frac{1}{A^2} & \text{si} -\frac{1}{A} \leq x, y \leq \frac{1}{A} \\ 0 & \text{ailleurs.} \end{cases} \tag{21}$$

Sous l'hypothèse que la transformation τ_k est localement linéaire, la FEP transformée est un parallélogramme uniformément distribué. Dans ce cas, nous pouvons faire un hypothèse plus forte encore en supposant que la transformation conserve les aires des pixels, ce qui est faux en réalité mais simplifie largement l'implémentation. On prendra par la suite une aire de pixel unitaire. Parce que la couleur est constante et égale $u(\mathrm{p})$ sur toute l'aire du pixel p dans la vue cible, la convolution précédente peut s'écrire comme dans [15] :

$$v_k(\mathrm{x}_m) = \sum_{\mathrm{p} \in \Gamma} u(\mathrm{p}) \int_{\mathrm{p}} b_k(\mathrm{x}' - \tau_k(\mathrm{x}_m))\, \mathrm{dx}', \tag{22}$$

donc la couleur du pixel m dans la vue source est

$$v_k(\mathrm{m}) = \sum_{\mathrm{p} \in \Gamma} B_{k,m,p} u(\mathrm{p}), \tag{23}$$

où $B_{k,m,p} = \int_{\mathrm{p}} b_k(\mathrm{x}' - \tau_k(\mathrm{x}_m))\, \mathrm{dx}'$ est l'aire d'intersection entre la projection du pixel dans la vue cible u et un pixel p de cette vue. Cela revient à faire une interpolation bilinéaire des intensités de u.

5.2 Le système linéaire

Pour chaque pixel m de chaque vue en entrée v_k nous obtenons une équation similaire à (23). Soit $\mathbf{V}^*$ le vecteur de tous les pixels de toutes les vues sources mis dans une seule grande colonne $(v_0(0), v_0(1), ..., v_{K-1}(M-1))^t$, $\mathbf{U}$ le vecteur colonne contenant la solution courante $(u(0), ..., u(N-1))^t$, et $\mathbf{B}$ la matrice $KM \times N$ qui contient les coefficients $B_{k,m,p}$. On peut donc écrire naturellement $\mathbf{V} = \mathbf{BU}$. Par conséquent on exprime l'énergie (9) comme un système linéaire :

$$E_{color}(\mathbf{U}) = (\mathbf{BU} - \mathbf{V}^*)^t \mathbf{W} (\mathbf{BU} - \mathbf{V}^*), \tag{24}$$

où $\mathbf{W}$ est une matrice $KM \times KM$ diagonale qui contient les poids $|J_{\mathrm{x}'}(\beta_k)|\omega_k$. Pour minimiser cette énergie nous dérivons le système linéaire, et obtenons les équations normales qui nous apportent un estimateur de la solution $\hat{\mathbf{U}}$:

$$\mathbf{B}^t \mathbf{W} \mathbf{B} \hat{\mathbf{U}} = \mathbf{B}^t \mathbf{W} \mathbf{V}^*. \tag{25}$$

La matrice $\mathbf{B}^t\mathbf{WB}$ n'est en général pas inversible. Le système linaire peut être résolu par n'importe quelle méthode des moindres carrés linéaires.
De même que le terme sur les couleurs de l'image, le terme portant sur les gradients est

$$E_{grad}(\mathbf{U}) = (\mathbf{B}\nabla\mathbf{U} - \nabla\mathbf{V}^*)^t (\mathbf{B}\nabla\mathbf{U} - \nabla\mathbf{V}^*) \tag{26}$$

et se dérive identiquement.

6 Expériences

6.1 Base de données structurées

Les premières expériences ont été réalisées à partir d'une base de données d'images *light-field* [34], prises du *HCI Light Field Benchmark Datasets* et de la *Stanford Light Field Archive*. Pour chaque base d'images nous utilisons une matrice de vues adjacentes (3×3 ou 1×9). Nous appliquons la méthode de Wanner *et al.* [32] pour estimer les disparités entre les vues et l'incertitude de la géométrie à partir des huit vues voisines. La vue centrale est rendue par chaque algorithme testé puis comparée avec l'image originale qui sert de référence pour évaluer les performances de l'algorithme.
α et λ sont fixés à leur valeur d'origine dans les expériences précédentes [26, 33], respectivement 1.0 et 0.1. Nous faisons varier γ de 0 à 3 pour observer l'influence du terme sur les gradients. Un γ nul est bien entendu équivalent à minimiser la même fonctionnelle que [26], mais notre implémentation diffère légèrement de la leur, ce qui explique pourquoi nous avons représenté les deux dans le tableau des résultats 1. Nous comparons également notre méthode à [33]. Toutes les expériences sont réalisées sur carte graphique (nVidia GTX Titan). La résolution du système prend entre 2 et 3 secondes pour des images de résolution 768×768.
Le PSNR (plus il est haut mieux c'est) et le $\mathrm{DSSIM} = 10^4(1 - \mathrm{SSIM})$ [31] (plus il est faible mieux c'est) sont calculés par rapport à la vue de référence pour évaluer nos résultats. Le deuxième jeu d'expériences a été réalisé avec les mêmes images, mais avec une géométrie planaire – la disparité estimée est nulle, et les transformation entre les

vues sont des identités. L'incertitude de la géométrie est augmentée.

Notre terme sur les gradients améliore systématiquement les résultats avec une disparité estimée, et très souvent pour une disparité planaire. La qualité des images générées est une fonction croissante de γ. Nous interprétons ceci par le fait que le terme sur les gradients ajoute de nouvelles contraintes au système, permettant à l'algorithme d'optimisation une meilleure convergence vers le minimum global de l'énergie.

6.2 Base de données non structurées

Les expériences suivantes (Figure 4) ont été réalisées sur des vues réelles prises de la base de données de Strecha [29], *fountain* et *herzjesu*. La reconstruction 3D est effectuée par la méthode décrite dans la section 3. Deux types de reconstructions ont été utilisée : une grossière mais plus complète, et une précise mais plus incomplète. Nous générons la vue centrale, et nous gardons l'originale pour l'évaluation.

Comme le système est faiblement contraint, des hautes fréquences apparaissent dans les zones visibles depuis peu de caméras. Ces artefacts sont accentués par une estimation très bruitée de la profondeur près des régions d'occultation (autour du poisson de la fontaine, ou de la gravure de Jésus. Le paramètre λ contrôlant le terme de régularisation *Total Variation* a été augmenté à 0.003 pour réduire l'apparition de ces hautes fréquences. Mais le résultat est peu convainquant car les images perdent alors du détail comparées aux originales. Nous avons alors baissé λ pour conserver tous les traits de la gravure et ajouté le terme sur les gradients ($\gamma = 1.0$). Nous pouvons voir sur les images, quelle que soit la géométrie utilisée pour le rendu, que les artefacts disparaissent tout en préservant les détails de l'image. Le terme sur les couleurs est conservé pour que la couleur originale des images ne soit pas affectée mais mis à une valeur très faible ($\alpha = 0.01$). L'ajout du terme sur les gradients a en outre permis d'empêcher l'apparition de faux contours près des frontières de visibilité, garantissant ainsi la propriété de *continuité*.

7 Discussion et conclusion

Nous avons présenté une méthode de rendu basé image qui permet de générer une nouvelle vue à partir d'un ensemble générique et non structuré d'images. Cette méthode est inspiré par les travaux de Pujades *et al*. [26], qui ont oeuvré pour formaliser la plupart des « propriétés désirables » listées dans l'article phare de Buehler *et al*. [4]. Leur approche fut d'introduire une formulation bayésienne du problème de rendu et d'obtenir la vue cherchée par un processus d'optimisation. La seule propriété qu'ils n'ont pu formaliser fut la propriété de *continuité*, qui énonce que les contributions de chaque vue source doivent être des fonctions continues des coordonnées des pixels.

Nous avons montré qu'un moyen de garantir cette *continuité* est de déclarer des les contours, textures et détails ne devraient pas être créés dans l'image cible s'ils ne sont pas présents dans les images sources aux endroits visibles. Cela implique l'ajout d'un terme additionnel portant sur les données sources, basé sur les gradients des images. L'énergie ainsi modifiée peut être minimisée en résolvant de façon itérative un système linéaire dérivé de la fonctionnelle. Ce système est alors plus contraint et mieux conditionné que le précédent, ce qui empêche l'apparition d'artefacts près des frontières de visibilité.

Ce résultat montre une nette amélioration par rapport aux précédentes méthodes de rendu basées sur les intensités, à la fois en terme de mesure qualitative et en terme de qualité subjective.

Cette méthode pourrait être retravaillée pour optimiser directement les gradients de la vue cible, plutôt que les intensités ; puis l'intensité de l'image pourrait être reconstruite en résolvant l'équation de Poisson, comme il es fait dans [18]. Cela devrait complètement enlever toutes les variations dans l'image synthétisée qui viennent de discontinuités des fonctions de visibilité, qui sont toujours visibles dans nos résultats, bien qu'atténuées.

Références

[1] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 294–302, New York, NY, USA, 2004. ACM.

[2] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9) :1167–1183, Sept. 2002.

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009*, pages 693–696, Apr. 2009.

[4] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen. Unstructured lumigraph rendering. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 425–432, New York, NY, USA, 2001. ACM.

[5] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2) :89–97, Jan. 2004.

[6] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis. Depth synthesis and local warps for plausible image-based navigation. *ACM Trans. Graph.*, 32(3) :30 :1–30 :12, July 2013.

[7] T. S. Cho, C. Zitnick, N. Joshi, S. B. Kang, R. Szeliski, and W. Freeman. Image restoration by matching gradient distributions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(4) :683–694, April 2012.

[8] A. Davis, M. Levoy, and F. Durand. Unstructured light fields. *Computer Graphics Forum*, 31(2pt1) :305–314, 2012.

[9] S. Fuhrmann and M. Goesele. Floating scale surface reconstruction. *ACM Transactions on Graphics (TOG)*, 33(4) :46, 2014.

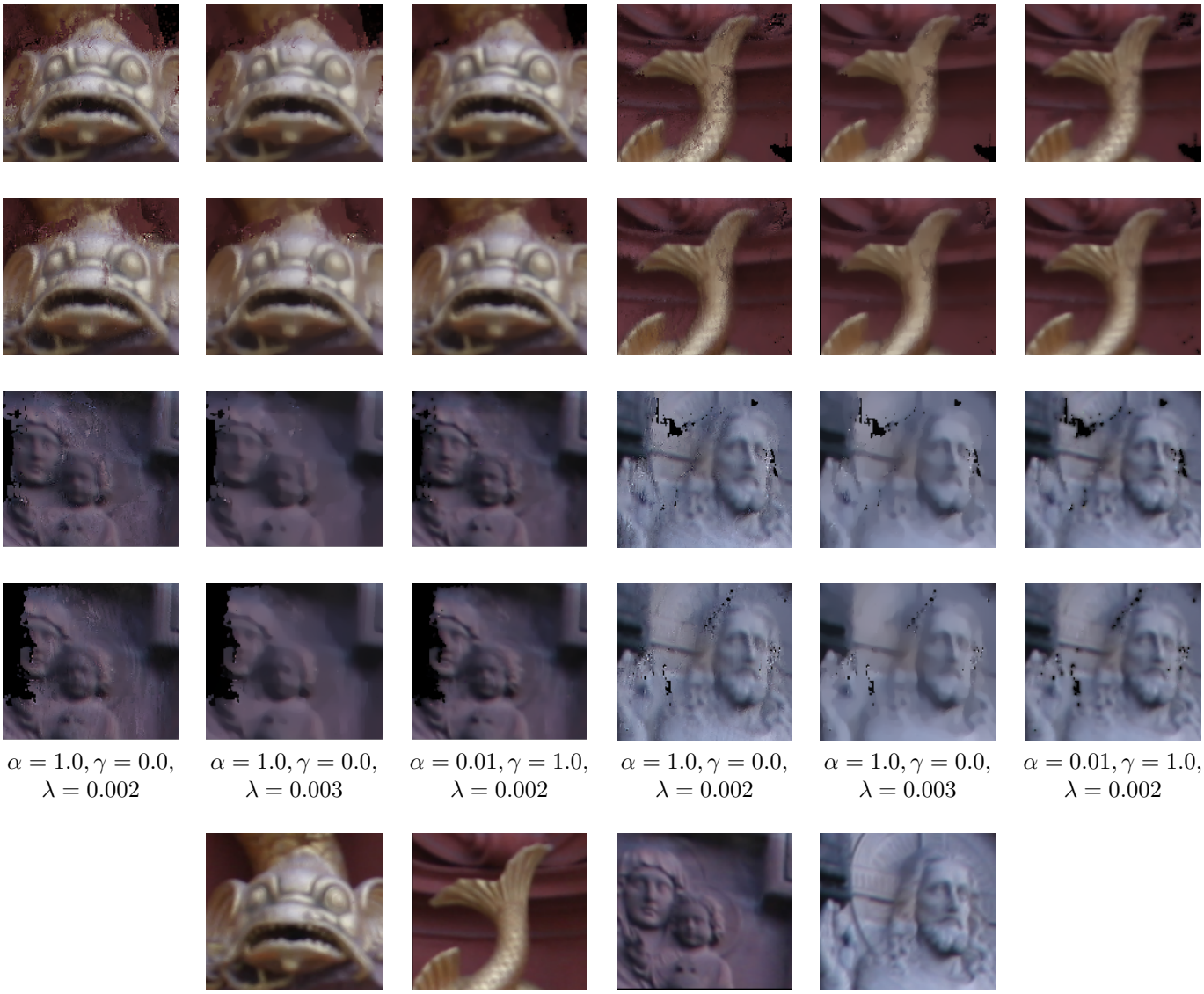

FIGURE 4 – Rendu avec différents paramètres. Chaque colonne montre les résultats sur les images *fountain* et *herzjesu* pour l'ensemble de paramètres $(\alpha, \gamma, \lambda)$ qui contrôlent la proportion des différents termes dans la formule de l'énergie. L'approche proposée est $\gamma \neq 0$. Première et troisième ligne : reconstruction de la géométrie à partir des images originelles. Deuxième et quatrième ligne : reconstruction de la géométrie à partir des images sous-échantillonnées $\times 4$. Dernière ligne : les images de référence pour comparaison.

	HCI light fields, raytraced				HCI light fields, gantry				Standford light fields, gantry					
	buddha		stillLife		maria		couple		truck		gum nuts		tarot	
Estimated disparity														
SAVSR [33]	42.84	17	30.13	58	40.06	53	26.55	226	33.75	408	31.82	1439	28.71	60
BVS [26]	42.37	18	30.45	55	40.10	53	28.50	178	33.78	**407**	31.93	1437	28.88	58
$\gamma = 0.0$	43.07	15	30.75	50	39.91	53	32.93	92	33.73	434	31.98	1430	29.37	51
$\gamma = 1.0$	43.28	14	30.83	49	40.14	51	33.05	90	33.82	430	32.08	1428	29.58	48
$\gamma = 2.0$	43.43	13	30.86	49	40.36	48	33.15	88	33.91	427	32.17	1426	29.74	46
$\gamma = 3.0$	**43.59**	**12**	**30.90**	**48**	**40.55**	**46**	**33.22**	**87**	**33.98**	423	**32.25**	**1424**	**29.89**	**44**
Planar disparity														
SAVSR [33]	34.28	74	21.28	430	31.65	144	20.07	725	32.48	419	30.55	1403	22.64	278
BVS [26]	37.51	44	22.24	380	34.38	99	**22.88**	**457**	33.79	386	31.30	1378	23.78	218
$\gamma = 0.0$	37.69	42	**22.27**	**377**	34.28	100	22.74	468	34.50	367	31.38	1359	24.47	189
$\gamma = 1.0$	37.74	41	**22.27**	378	34.34	97	22.74	468	34.57	365	31.39	1359	24.51	187
$\gamma = 2.0$	37.80	**40**	22.26	378	34.40	95	22.74	468	34.66	362	**31.43**	1358	24.50	187
$\gamma = 3.0$	**37.84**	**40**	22.25	378	**34.45**	**93**	22.74	468	**34.68**	**360**	31.42	**1357**	**24.58**	**185**

TABLE 1 – Résultats numériques sur les bases d'images synthétiques et réelles [34]. Notre méthode est comparée à celle de Wanner *et al*. [33] et de Pujades *et al*. [26]. Le *proxy géométrique* est soit estimé par [32] soit mis à profondeur constante avec une grande incertitude. Pour chaque *light-field*, la première valeur est le PSNR (plus il est aussi mieux c'est), la seconde est 10^{-4} fois le DSSIM. $\mathrm{DSSIM} = 10^4(1 - \mathrm{SSIM})$ [31] (plus il est faible mieux c'est). La meilleure performance est en gras. Voir la section 6.2 pour de plus amples détails sur l'expérience.

[10] S. Fuhrmann, F. Langguth, and M. Goesele. MVE – a multiview reconstruction environment. In *Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage (GCH)*, volume 6, page 8, 2014.

[11] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8) :1362–1376, Aug. 2010.

[12] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[13] B. Goldluecke and D. Cremers. Superresolution texture maps for multiview reconstruction. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1677–1684, Sept. 2009.

[14] B. Goldluecke and D. Cremers. An approach to vectorial total variation based on geometric measure theory. In *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 327–333, June 2010.

[15] R. Hardie, K. Barnard, and E. Armstrong. Joint MAP registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Transactions on Image Processing*, 6(12) :1621–1633, Dec. 1997.

[16] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, pages 61–70, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[17] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. In *ACM SIGGRAPH 2007 Papers*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.

[18] J. Kopf, F. Langguth, D. Scharstein, R. Szeliski, and M. Goesele. Image-based rendering in the gradient domain. *ACM Trans. Graph.*, 32(6) :199 :1–199 :9, Nov. 2013.

[19] A. Levin, A. Zomet, and Y. Weiss. Learning how to inpaint from global image statistics. In *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, pages 305–312 vol.1, Oct. 2003.

[20] C. Lipski, F. Klose, and M. Magnor. Correspondence and depth-image based rendering a hybrid approach for free-viewpoint video. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(6) :942–951, June 2014.

[21] C. Lipski, C. Linz, K. Berger, A. Sellent, and M. Magnor. Virtual video camera : Image-based viewpoint navigation through space and time. *Computer Graphics Forum*, 29(8) :2555–2568, 2010.

[22] J. McCann and N. S. Pollard. Real-time gradient-domain painting. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 93 :1–93 :7, New York, NY, USA, 2008. ACM.

[23] P. Moulon, P. Monasse, and R. Marlet. La bibliothèque openMVG : open source multiple view geometry. In *Orasis, Congrès des jeunes chercheurs en vision par ordinateur*, 2013.

[24] R. Ortiz-Cayon, A. Djelouah, and G. Drettakis. A bayesian approach for selective image-based rendering using superpixels. In *3D Vision (3DV), International Conference on*. IEEE, 2015.

[25] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, pages 313–318, New York, NY, USA, 2003. ACM.

[26] S. Pujades, F. Devernay, and B. Goldluecke. Bayesian view synthesis and image-based rendering principles. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3906–3913, June 2014.

[27] H.-Y. Shum, S.-C. Chan, and S. B. Kang. *Image-based rendering*. Springer Science & Business Media, 2008.

[28] S. N. Sinha, J. Kopf, M. Goesele, D. Scharstein, and R. Szeliski. Image-based rendering for scenes with reflections. *ACM Trans. Graph.*, 31(4) :100, 2012.

[29] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, pages 1–8.

[30] J. Sun, J. Sun, Z. Xu, and H.-Y. Shum. Gradient profile prior and its applications in image super-resolution and

enhancement. *Image Processing, IEEE Transactions on*, 20(6) :1529–1542, June 2011.

[31] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment : from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4) :600–612, April 2004.

[32] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4D light fields. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 41–48, June 2012.

[33] S. Wanner and B. Goldluecke. Spatial and angular variational super-resolution of 4D light fields. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *Computer Vision – ECCV 2012*, number 7576 in Lecture Notes in Computer Science, pages 608–621. Springer Berlin Heidelberg, 2012.

[34] S. Wanner, S. Meister, and B. Goldluecke. Datasets and benchmarks for densely sampled 4D light fields. In *Vision, Modelling and Visualization (VMV)*. 2013.

[35] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 600–608, New York, NY, USA, 2004. ACM.

[36] A. Zomet, A. Levin, S. Peleg, and Y. Weiss. Seamless image stitching by minimizing false edges. *IEEE Transactions on Image Processing*, 15(4) :969–977, Apr. 2006.